WHAT IS CLAIMED IS:

(Currently Amended) 1. A method of simulating physical dynamics of a predetermined set of objects that are part of a computer/video game, the objects connected to each other at one or more respective links, with at least one link representing a hard contact between separate objects, wherein the method utilizes a game system comprising a collision subsystem, a dynamics subsystem, a game logic subsystem, a graphics subsystem and one or more central processing units supporting the game system; the method further comprising:

a. grouping a first and a second object in the predetermined set of objects to define a first binary object;

b. solving for the physical dynamics of the objects in the first binary object at a first set of links;

c. grouping a third object to the first binary object to define a second binary object, the third object having at least one link to the first binary object, thereby defining a second set of links;

d. solving a solution for the physical dynamics of the objects in the second binary object at the second set of links; and

e. recursively grouping additional objects to create additional binary objects and solving for the physical dynamics of the additional binary objects.

(Original) 2. The method of claim 1 including the step of providing, for each link, one or more link weight values operable to constrain the solution.

(Currently Amended) 3. The method of claim 2 further including the step of performing an iterative solution method multiple times where at least one link weight

value is adjusted at each iteration.

(Original) 4. The method of claim 3 where the link weight values are adjusted to maintain a set of constraints on the links within a predetermined acceptable tolerance.

(Original) 5. The method of claim 4 where the set of constraints includes the following constraints: the objects cannot interpenetrate each other and no adhesive normal forces are applied at the links.

(Original) 6. The method of claim 5 where the predetermined acceptable tolerance includes a predetermined amount of interpenetration at a link.

(Original) 7. The method of claim 6 where the predetermined acceptable tolerance includes a predetermined amount of adhesive normal force at a link.

(Currently Amended) 8. The method of claim 7 where the set of constraints further includes the constraint that, at a respective link, either the relative lateral motion is zero or the friction force at the link is equal to the normal force times a coefficient of friction.

(Original) 9. The method of claim 8 where the predetermined acceptable tolerance includes a predetermined difference between the friction force at a link and the normal force times the coefficient of friction.

(Currently Amended) 10. A method of simulating physical dynamics of a predetermined set of objects that are part of a computer video game, the objects connected to each other by at least one respective link, and where at least one object is not a rigid body, wherein the method utilizes a game system comprising a collision subsystem, a dynamics subsystem, a game logic subsystem, a graphics subsystem and one

or more central processing units supporting the game system; the method further comprising:

a. providing, for at least one object, a set of reaction values describing the motion of the object in response to applied forces;

b. solving for the physical dynamics of the set of objects using the reaction values;

c. changing the reaction values in response to force for at least one object to provide a set of adjusted reaction values;

d. solving for the physical dynamics of the objects using the set of adjusted reaction values; and

e. repeating steps c and e until the solution of step d is within a predetermined acceptable tolerance.

(Original) 11. The method of claim 10 further including the step of creating a nested grouping of a plurality of binary objects from the objects in the set.

(Currently Amended) 12. The method of claim 11 where the step of solving a solution for the physical dynamics of the objects includes the step of starting with the most deeply nested binary object and proceeding outward, solving for the physical dynamics of the objects in the binary objects at the respective links.

(Original) 13. The method of claim 12 further including the step of providing, for each link, one or more link weight values operable to constrain the solution.

(Original) 14. The method of claim 13 where the step of changing the reaction values for at least one object further includes the step of adjusting at least one link weight value.

(Original) 15. The method of claim 14 where the link weight values are adjusted to maintain a set of constraints on the links within a predetermined acceptable tolerance.

(Currently Amended) 16. A method of simulating physical dynamics of a predetermined set of objects that are part of a computer/video game, the objects connected to each other at one or more respective links, with at least one link representing a hard contact between separate objects, wherein the method utilizes a game system comprising a collision subsystem, a dynamics subsystem, a game logic subsystem, a graphics subsystem and one or more central processing units supporting the game system; the method further comprising:

a. grouping the objects in the predetermined set of objects into two binary objects to define a first binary object and a second binary object;

b. grouping the objects in the first binary object into a subgroup of binary objects to define a nested group of binary objects in the first binary object;

c. grouping the objects in the second binary object into a subgroup of binary objects to define a nested group of binary objects in the second-binary object; and

d. starting with the most deeply nested binary object and proceeding outward, solving a solution for the physical dynamics of the objects in the binary objects at the respective links.

(Original) 17. The method of claim 16 including the step of providing, for each link, one or more link weight values operable to constrain the solution.

(Currently Amended) 18. The method of claim 17 further including the step of performing an iterative solution method multiple times where at least one link weight

value is adjusted at each iteration.

(Original) 19. The method of claim 18 where the link weight values are adjusted to maintain a set of constraints on the links within a predetermined acceptable tolerance.

(Original) 20. The method of claim 19 where the set of constraints includes the following constraints: the objects cannot interpenetrate each other and no adhesive normal forces are applied at the links.

(Original) 21. The method of claim 20 where the predetermined acceptable tolerance includes a predetermined amount of interpenetration at a link.

(Original) 22. The method of claim 21 where the predetermined acceptable tolerance includes a predetermined amount of adhesive normal force at a link.

(Currently Amended) 23. The method of claim 22 where the set of constraints further includes the constraint that, at a respective link, either the relative lateral motion is zero or the friction force at the link is equal to the normal force times a coefficient of friction.

(Original) 24. The method of claim 23 where the predetermined acceptable tolerance includes a predetermined difference between the friction force at a link and the normal force times the coefficient of friction.

(Currently Amended) 25. A method of simulating the physical dynamics of a predetermined set of objects that are part of a computer/video game, the objects connected to each other at one or more links, at least one object represented by a plurality of polygons, wherein the method utilizes a game system comprising a collision subsystem, a dynamics subsystem, a game logic subsystem, a graphics subsystem and one

or more central processing units supporting the game system; the method further comprising:

a. creating a nested grouping of a plurality of binary objects from the objects in the set, at least one binary object containing two or more links; and

b. starting with the most deeply nested binary object and proceeding outward, solving for the physical dynamics of the objects in the binary objects at said one or more links.

(Original) 26. The method of claim 25 where the solution maintains a set of constraints on the links within a predetermined acceptable tolerance.

(Original) 27. The method of claim 26 where the set of constraints includes the following constraints: the objects cannot interpenetrate each other and no adhesive normal forces are applied at the links.

(Original) 28. The method of claim 27 where the predetermined acceptable tolerance includes a predetermined amount of interpenetration at a link.

(Original) 29. The method of claim 28 where the predetermined acceptable tolerance includes a predetermined amount of adhesive normal force at a link.

(Currently Amended) 30. The method of claim 29 where the set of constraints further includes the constraint that, at a respective link, either the relative lateral motion is zero or the friction force at the link is equal to the normal force times a coefficient of friction.

(Original) 31. The method of claim 30 where the predetermined acceptable tolerance includes a predetermined difference between the friction force at a link and the normal force times the coefficient of friction.

(Original) 32. The method of claim 31 including the step of

providing, for each link, one or more link weight values operable to constrain the solution.

(Currently Amended) 33. The method of claim 32 further including the step of performing an iterative solution method multiple times where at least one link weight value is adjusted at each iteration.

(Currently Amended) 34. A method of simulating the physical dynamics of a predetermined set of objects that are part of a video game, the objects connected to each other at at least one respective link, wherein the method utilizes a game system comprising a collision subsystem, a dynamics subsystem, a game logic subsystem, a graphics subsystem and one or more central processing units supporting the game system; the method further comprising:

a. providing a set of equations that when solved define a solution to the physical dynamics of the predetermined set of objects, the solution having the following constraints: the objects cannot interpenetrate each other and no adhesive normal forces can be applied at the links;

b. assigning at least one link weight to each of the links in the predetermined set of objects;

c. solving for the physical dynamics of the objects using the assigned weights using an iterative solution method;

d. adjusting the assigned link weights if the constraints are violated at a link;

e. solving an iterative solution for the physical dynamics of the objects using the adjusted weights; and

f. repeating steps d. and e. until a solution is within a predetermined acceptable tolerance.

(Currently Amended) 35. The method of claim 34 wherein a predetermined acceptable tolerance includes a predetermined amount of adhesive normal force at a link.

(Original) 36. The method of claim 35 wherein the predetermined acceptable tolerance includes a predetermined amount of interpenetration between two objects at a link.

(Original) 37. The method of claim 36 wherein the weights are decreased for links where adhesive normal force is applied.

(Original) 38. The method of claim 37 wherein the weights are increased for links where interpenetration occurs.

(Currently Amended) 39. A method of simulating the physical dynamics of a predetermined set of objects that are part of a video game, the objects connected to each other at one or more respective links, wherein the method utilizes a game system comprising a collision subsystem, a dynamic subsystem, a game logic subsystem, a graphics subsystem and one or more central processing units supporting the game system; the method further comprising:

a. providing a set of equations that when solved define a solution to the physical dynamics of the predetermined set of objects, the solution having the following constraints: the objects cannot interpenetrate each other and no adhesive normal forces can be applied at the links, and that, at a respective link, either the relative lateral velocity is zero, or the friction force is equal to the normal force at the link times a coefficient of friction;

b. assigning at least one link weight to each of the links in the predetermined set of objects;

c. solving for the physical dynamics of the objects using the assigned weights using an iterative solution method;

d. adjusting the link weights assigned to the links if the constraints are violated at a link;

e. solving an iterative solution for the physical dynamics of the objects using the adjusted weights; and

f. repeating steps d. and e. until a solution is within a predetermined acceptable tolerance.

(Original) 40. The method of claim 39 wherein a predetermined acceptable tolerance includes a predetermined amount of adhesive normal force at a link.

(Original) 41. The method of claim 40 wherein the predetermined acceptable tolerance includes a predetermined amount of interpenetration between two objects at a link.

(Currently Amended) 42. The method of claim 41 where the predetermined acceptable tolerance includes a predetermined difference between the friction force at a link and the normal force times a coefficient of friction.

(Original) 43. The method of claim 42 wherein the weights are decreased for links where adhesive normal force is applied.

(Original) 44. The method of claim 43 wherein the weights are increased for links where interpenetration occurs.

(Currently Amended) 45. A system for simulating the physical dynamics of a

set of objects within a video game, the objects connected to each other at one or more respective links, the system comprising:

a. a binary division unit having logic operable to create a nested grouping of a plurality of binary objects from the objects in the set;

b. a dynamics unit having logic operable to solve a set of physical dynamics equations; and

c. one or more central processing units supporting the system.

(Original) 46. The system of claim 45 where the dynamics unit comprises a set of multiple processors, each processor operable to solve a set of physical dynamics equations.

(Original) 47. The system of claim 46 where said multiple processors are used to solve the dynamics equations of multiple binary objects in parallel.

(Currently Amended) 48. The system of claim 47 where each link includes one or more link weight values operable to constrain a solution.

(Currently Amended) 49. The system of claim 48 where the dynamics unit further comprises logic operable to perform an iterative solution method multiple times wherein one or more link weight values are adjusted at each iteration.

(Original) 50. The system of claim 49 where the link weight values are adjusted to maintain a set of constraints for each link within a predetermined tolerance.

(Original) 51. The system of claim 50 where said set of constraints includes the following constraints: the objects cannot interpenetrate each other and no adhesive normal force is applied.

(Original) 52. The system of claim 51 where the set of constraints

further includes the constraint that, at a respective link, either the relative lateral motion between the objects is zero or the friction force at the link is equal to the normal force multiplied by a coefficient of friction.